

Гибридный эволюционный алгоритм для
одной задачи составления производственного
расписания

Борисовский П.А.

Институт математики им. С.Л. Соболева СО РАН (Омский филиал)

Исследование выполнено за счет гранта Российского научного фонда
№ 22-71-10015-П, <https://rscf.ru/project/22-71-10015-П/>.

Постановка задачи

$S = \{1, 2, \dots, s\}$ – продукты;

$U = \{1, 2, \dots, m\}$ – устройства;

$I = \{1, 2, \dots, n\}$ – операции;

D_s - спрос на продукт s ;

Для каждой операции i задано:

u_i – требуемое устройство,

s_i – производимый продукт,

r_i – производительность (тонн/час),

s_{ij} – время переналадки.

Построить расписание выпуска продукции в нужном объёме, минимизируя C_{\max} – момент завершения последней операции.

Модель ЧЦЛП

Бинарные переменные

x_{ik} такие, что $x_{ik} = 1$, если операция i на устройстве u_i занимает k -е по счету место.

Вещественные неотрицательные переменные:

δ_i – длительность операции i .

α_{uk} – длительность переналадки на устройстве u после k -й по счету операции.

Модель ЧЦЛП

$$\min C_{\max}$$

$$\sum_{i \in I} x_{ik} \leq 1, \quad k \in K;$$

$$\sum_{k \in K} x_{ik} \leq 1, \quad i \in I;$$

$$\sum_{i: s_i = s} r_i \delta_i \geq D_s, \quad s \in S;$$

$$\sum_{i \in I_u} x_{i,k-1} \geq \sum_{i \in I_u} x_{i,k}, \quad u \in U, k \in K, k > 1;$$

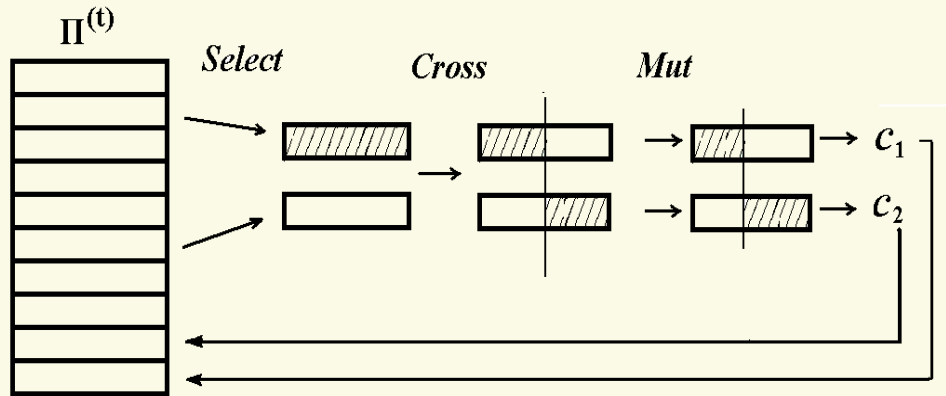
$$\alpha_{uk} \geq \sum_{j \in I_u} s_{ji} x_{j,k-1} - M(1 - x_{ik}), \quad u \in U, i \in I_u, k \in K, k > 1;$$

$$\sum_{i \in I_u} \delta_i + \sum_{k \in K} \alpha_{uk} \leq C_{\max}, \quad u \in U;$$

$$T_i^{\min} \sum_{k \in K} x_{ik} \leq \delta_i \leq T_i^{\max} \sum_{k \in K} x_{ik}, \quad i \in I;$$

$$x_{ik} \in \{0, 1\}, \delta_i, \alpha_{uk} \geq 0, \quad i \in I, u \in U, s \in S, k \in K.$$

Предыдущие исследования: Генетический алгоритм



1. Случайным образом построить начальную популяцию.
2. Пока не выполнен критерий останова, выполнять:
 - 2.1. Выбрать из популяции две родительские особи p_1 и p_2 .
 - 2.2. Применяя оператор скрещивания, построить двух потомков c_1 и c_2 .
 - 2.3. Применить к c_1 и c_2 оператор мутации.
 - 2.4. Удалить две худшие особи и поместить на их место c_1 и c_2 .

Представление решений

$\Pi = (i_1, i_2, \dots, i_l)$ - перестановка операций;

(L_1, L_2, \dots, L_s) - максимальное количество операций для производства соответствующего продукта.

Пример:

task	1	2	3	4	5	6	7	8	9
product	1	2	3	1	2	1	2	2	3
unit	1	1	2	1	2	2	2	1	2

Пусть дана перестановка $\pi = (6, 7, 5, 4, 8, 3, 9, 1, 2)$

и вектор $L = (2, 2, 1)$.

При декодировании получается назначение операций на устройства:

unit 1: 4 3

unit 2: 6 7 5

Вычисление целевой функции

После фиксации бинарных переменных требуется решить задачу ЛП

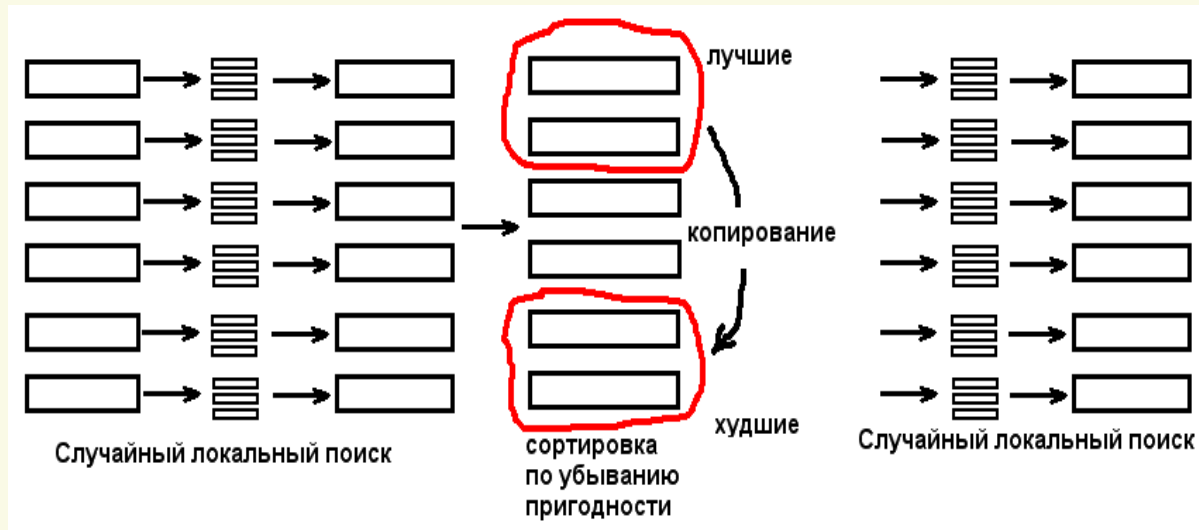
$$\text{Min } C_{\max}; \quad (1)$$

$$\sum_{i:s_i=s} r_i \delta_i \geq D_s, \quad s \in S; \quad (2)$$

$$\sum_{i \in I_u} \delta_i + \sum_{k \in K_u} \alpha_{uk} \leq C_{\max}, \quad u \in U; \quad (3)$$

$$\delta_i \geq 0, \quad i \in I, \quad C_{\max} \geq 0. \quad (4)$$

Алгоритм «иди с победителями» с итеративным локальным поиском (ILS-GWW) на GPU



Aldous, D., Vazirani, U.: "Go with the winners" Algorithms. Proceedings of 35th IEEE Symposium on Foundations of Computer Science (FOCS), 1994, pp. 492–501.

Борисовский П.А. Параллельный алгоритм «иди с победителями» для некоторых задач составления расписаний // Дискретный анализ и исследование операций. 2023. Т.30. №4. С.5–23.

Borisovsky P.A., Zakharov A.O., Zakharova Y.V. Evolutionary Algorithms for Customer Order Scheduling // Известия Иркутского государственного университета. 2025. Том.53. С.3-17.

Упрощенный алгоритм ADMM для задач ЛП

Настраиваемые параметры: $\rho \approx 0.1$, $\mathbf{T} \approx 10..30$,

$$s \approx 0.1, \mathbf{K} \approx 1/s.$$

$$x = (1, \dots, 1), \quad z = (1, \dots, 1) \quad u = (0, \dots, 0)$$

For $t = 1$ to \mathbf{T} :

$$x := z - u - c/\rho$$

For $k = 1$ to \mathbf{K} :

For $j = 1$ to m :

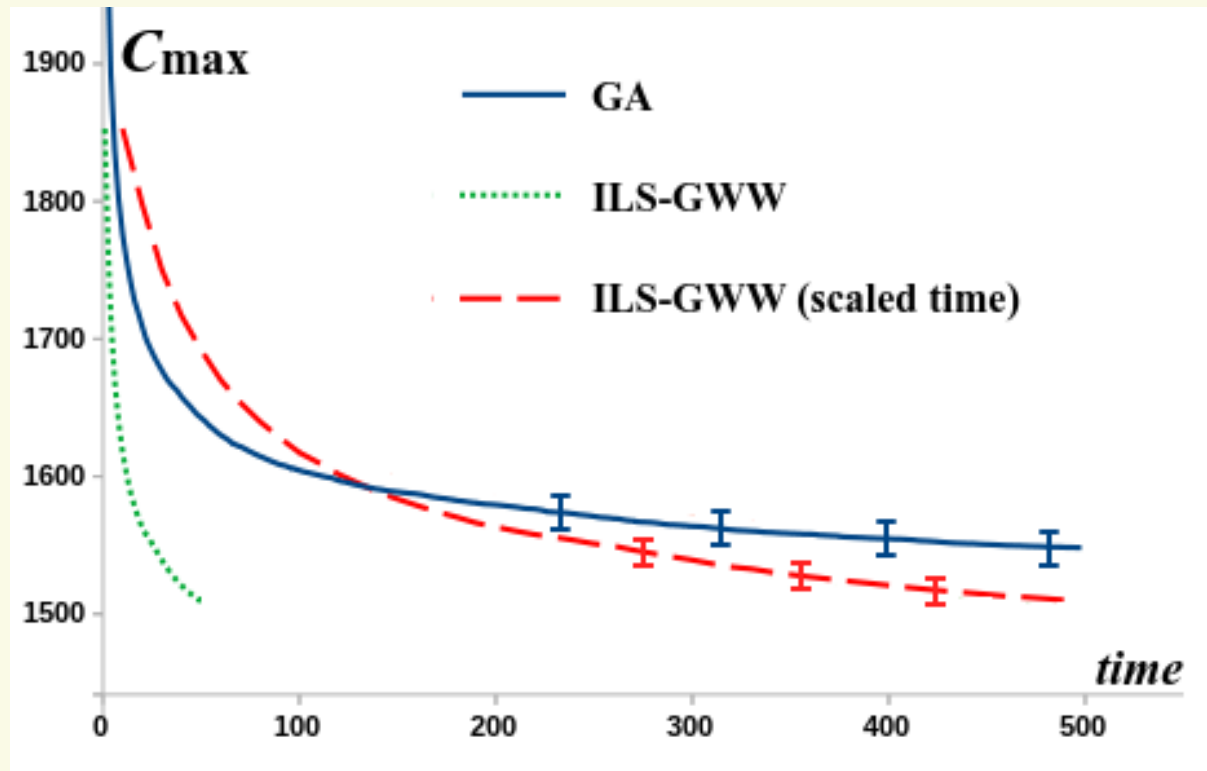
If $A_j x > b_j$:

$$x := x - \mathbf{s}(A_j x - b_j)A_j / (A_j \cdot A_j)$$

$$z := \max(0, x + u)$$

$$u := u + x - z$$

Сравнение алгоритмов



Динамическое программирование для задачи коммивояжера (Held, Karp, 1962)

Для подмножества городов P и города $i \notin P$ определим $f(i, P)$ - длина оптимального маршрута по городам $\{i\} \cup P$, начиная с i

Уравнения Беллмана

$$f(i, \{j\}) = a_{ij}$$

$$f(i, P) = \min_{j \in P} \{a_{ij} + f(j, P \setminus \{j\})\}.$$

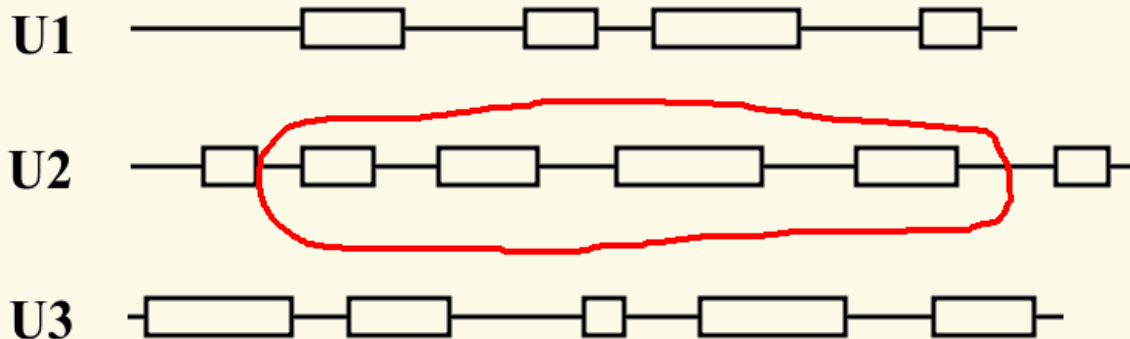
Динамическое программирование для задачи коммивояжера

Время работы алгоритма ДП

<i>n</i>	CPU AMD Phenom II (2008 г.)	CPU AMD EPYC 7502 (2019 г.)	GPU GTS 450 (2010 г.)	GPU Tesla V100 (2017 г.)
24	24 s.	13.5 s.	0.75 s.	14 ms.
25	53 s.	30 s.	1.5 s.	29 ms.
26	120 s.	71 s.	-	65 ms.
27	267 s.	153 s.	-	140 ms.

Гибридная метаэвристика

Для некоторого решения в ходе работы метаэвристики выбрать небольшую подпоследовательность операций и применить ДП для оптимизации переналадок.



Результаты экспериментов

#	S	U	I	CPLEX		GA		GA-DP	
				Cmax	ch.time	Cmax	ch.time	Cmax	ch.time
1	30	2	80	297	76	288	59	281	48
2	40	2	150	334	143	307	88	291	57
3	50	3	180	274	192	251	129	234	75
4	60	3	200	327	181	304	114	295	92
5	150	10	500	311	675	301	539	281	290
6	300	10	1000	–	–	603	2130	502	1026

Спасибо за внимание!